IN THE CLAIMS

Please amend the claims as follows.

For the Examiner's convenience, a list of all claims is included below.

1. (Currently amended) A method comprising:

accessing a reference array, the reference array referencing at least one data object, each of the at least one data object having a contents stored in a corresponding memory location;

determining a new memory location for the contents of each of the at least one data object; and

copying the contents of the at least one data object directly to the new memory location thus creating a new data object for each of the at least one data object, each new data object having a new data object contents, the contents of the at least one data object copied using a non-temporal streaming store, such that upon copying the contents of the at least one data object to the new memory location, the contents of each new data object does not get stored to a cache memory.

2. (Original) The method of claim 1 wherein the contents of consecutively referenced data objects are copied to consecutive memory locations.

3. (Previously presented) The method of claim 2 wherein copying further includes copying the contents of the at least one data object using a write combine operation.

4. (Original) The method of claim 3 implemented upon a computing system having a central processing unit wherein an amount of data copied depends upon central processing unit parameters.

5. (Original) The method of claim 4 wherein the computing system operates in a dynamic run-time environment.

6. (Original) The method of claim 5 wherein the run-time environment is selected from the group consisting of JAVA and CLI.

7. (Original) The method of claim 6 implemented as the copy phase of a moving garbage collection algorithm.

8. (Currently amended) A machine-readable medium that provides executable instructions, which when executed by a processor, cause the processor to perform a method, the method comprising:

accessing a reference array, the reference array referencing at least one data object, each of the at least one data object having a contents stored in a corresponding memory location;

determining a new memory location for the contents of each of the at least one data object; and

copying the contents of the at least one data object directly to the new memory location thus creating a new data object for each of the at least one data object, each new data object having a new data object contents, the contents of the at least one data object copied using a non-temporal streaming store, such that upon copying the contents of the at least one data object to

the new memory location, the contents of each new data object does not get stored to a cache memory.

9. (Original) The machine-readable medium of claim 8 wherein the contents of consecutively referenced data objects are copied to consecutive memory locations.

10. (Previously presented) The machine-readable medium of claim 9 wherein copying further includes copying the contents of the at least one data object using a write combine operation.

11. (Original) The machine-readable medium of claim 10 implemented upon a computing system having a central processing unit wherein an amount of data copied depends upon central processing unit parameters.

12. (Original) The machine-readable medium of claim 11 wherein the computing system operates in a dynamic run-time environment.

13. (Original) The machine-readable medium of claim 12 wherein the run-time environment is selected from the group consisting of JAVA and CLI.

14. (Original) The machine-readable medium of claim 13 implemented as the copy phase of a moving garbage collection algorithm.

15. (Currently amended) An apparatus comprising:

a register to hold a reference array, the reference array referencing at least one data object, each of the at least one data object having a contents;

a memory region corresponding to each data object to hold the contents of each of the at least one data object; and

a central processing unit to determine a new memory location for the contents of each of the at least one data object, and copy the contents of the at least one data object directly to the new memory location thus creating a new data object for each of the at least one data object, each new data object having a new data object contents, the contents of the at least one data object copied using a non-temporal streaming store, such that upon copying the contents of the at least one data object to the new memory location, the contents of each new data object does not get stored to a cache memory.

16. (Previously Presented) The apparatus of claim 15 wherein the contents of consecutively referenced data objects are copied to consecutive memory locations.

17. (Previously presented) The apparatus of claim 16 wherein copying further includes copying the contents of the at least one data object using a write combine operation.

18. (Original) The apparatus of claim 17 implemented upon a computing system having a central processing unit wherein an amount of data copied depends upon central processing unit parameters.

19. (Original) The apparatus of claim 18 wherein the computing system operates in a dynamic run-time environment.

20. (Original) The apparatus of claim 19 wherein the run-time environment is selected from the group consisting of JAVA and CLI.

21. (Original) The apparatus of claim 20 implemented as the copy phase of a moving garbage collection algorithm.

22. (Currently amended) A method comprising:

determining a data object to be a live data object;

determining a new memory location for the data object;

updating references associated with the data object; and

using a non-temporal streaming store to copy a contents of the data object directly to the new memory location, thus creating a new data object corresponding to the data object, such that a copy of a contents of the new data object is not stored to a cache memory.